

11/25/98
JCS511 U.S. PTO

A
JCS511 U.S. PTO
09/19/96
11/25/98

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of: David A. Sobeski et al.

Title: DYNAMIC OBJECT BEHAVIOR FOR OBJECT-ORIENTED COMPUTING ENVIRONMENTS

Attorney Docket No.: 777.172US1

PATENT APPLICATION TRANSMITTAL

BOX PATENT APPLICATION

Assistant Commissioner for Patents
Washington, D.C. 20231

We are transmitting herewith the following attached items and information (as indicated with an "X"):

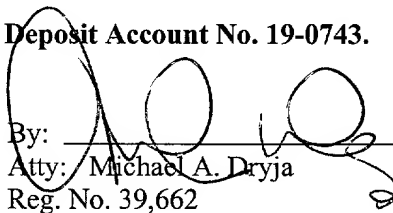
- X Utility Patent Application under 37 CFR § 1.53(b) comprising:
X Specification (13 pgs, including claims numbered 1 through 19 and a 1 page Abstract).
X Formal Drawing(s) (3 sheets).
X Signed Combined Declaration and Power of Attorney (3 pgs).
X Check in the amount of \$916.00 to pay the filing fee.
X Assignment of the invention to Microsoft Corporation (2 pgs) and Recordation Form Cover Sheet.
X Check in the amount of \$40.00 to pay the Assignment recording fee.
X Return postcard.

The filing fee has been calculated below as follows:

	No. Filed	No. Extra	Rate	Fee
TOTAL CLAIMS	19 - 20 =	0	x 18 =	\$0.00
INDEPENDENT CLAIMS	5 - 3 =	2	x 78 =	\$156.00
] MULTIPLE DEPENDENT CLAIMS PRESENTED				\$0.00
BASIC FEE				\$760.00
TOTAL				\$916.00

Please charge any additional required fees or credit overpayment to Deposit Account No. 19-0743.

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
P.O. Box 2938, Minneapolis, MN 55402 (612-373-6900)

By: 
Atty: Michael A. Dryja
Reg. No. 39,662

Customer Number 21186

"Express Mail" mailing label number: EL185283807US

Date of Deposit: November 25, 1998

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231.

By: Chris Hammond

Signature: 

DYNAMIC OBJECT BEHAVIOR FOR OBJECT-ORIENTED COMPUTING ENVIRONMENTS

RELATED APPLICATIONS

This application is related to the following applications, which are hereby incorporated by reference: "Object Model for Object-Oriented Computing Environments," [SLWK docket 777.169US1], "Dynamic Data Cache for Object-Oriented Computing Environments" [SLWK docket 777.171US1].

FIELD OF THE INVENTION

This invention relates generally to object-oriented computing environments, and more particularly to dynamic object behavior for such environments.

BACKGROUND OF THE INVENTION

Object-oriented programming environments are currently the standard environment in which computer programs are developed. For example, within the Microsoft Windows operating system, programs may be developed using the Component Object Model (COM) architecture. Object-oriented programming environments provide a modular manner by which developers can develop complex and sophisticated computer programs.

Generally, an object may include data and methods by which that data is accessed and changed. Thus, new methods may be added to the object for accessing and changing the data in other ways not previously possible, without affecting existing methods, and the external objects that rely on these existing methods. Upon receiving a message, or in response to an event, an object typically executes a particular method, resulting in the data within that object being retrieved or changed.

SUMMARY OF THE INVENTION

The invention provides for dynamic object behavior for object-oriented environments. In one embodiment, a system includes an object, a data store, and an application. The object has a plurality of dynamic behaviors. The data store stores data regarding the plurality of dynamic

behaviors. The application instantiates the object from data stored in the data store regarding the plurality of dynamic behaviors. In addition, of the behaviors available to the object, the object is able to instantiate only a sub-set of all the behaviors, at run-time, depending on, for example, the current environment in which the object is instantiated.

The dynamic object behavior of the invention provides for advantages not found in the prior art. For example, typically within the prior art, an object's behavior is permanently defined, making it difficult to change that object's behavior. However, in embodiments of the invention, the object's behavior may be changed, by changing the data regarding the plurality of behaviors in the data store (such as, adding a behavior or removing a behavior), or by having the object instantiate different of all of the possible behaviors at run-time, as a result of a dynamic determination made at run-time by the object. The invention includes systems, methods, computers, and computer-readable media of varying scope. Besides the embodiments, advantages and aspects of the invention described here, the invention also includes other embodiments, advantages and aspects, as will become apparent by reading and studying the drawings and the following description.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows a diagram of the hardware and operating environment in conjunction with which embodiments of the invention may be practiced;

FIG. 2 shows a block diagram of a system according to one embodiment of the invention; and,

FIG. 3 shows a flowchart illustrating a method according to one embodiment of the invention.

DETAILED DESCRIPTION

In the following detailed description of exemplary embodiments of the invention, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration specific exemplary embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical, electrical and other changes may be made without departing from the spirit or scope of the present invention. The following detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

The detailed description is divided into four sections. In the first section, the hardware and the operating environment in conjunction with which embodiments of the invention may be practiced are described. In the second section, a system of one embodiment of the invention is presented. In the third section, a method, in accordance with an embodiment of the invention, is provided. Finally, in the fourth section, a conclusion of the detailed description is provided.

Hardware and Operating Environment

Referring to FIG. 1, a diagram of the hardware and operating environment in conjunction with which embodiments of the invention may be practiced is shown. The description of FIG. 1 is intended to provide a brief, general description of suitable computer hardware and a suitable computing environment in conjunction with which the invention may be implemented. Although not required, the invention is described in the general context of computer-executable instructions, such as program modules, being executed by a computer, such as a personal computer. Generally, program modules include routines, programs, objects, components, data structures, etc., that perform particular tasks or implement particular abstract data types.

Moreover, those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, network PCS, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed

computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

The exemplary hardware and operating environment of FIG. 1 for implementing the invention includes a general purpose computing device in the form of a computer 20, including a processing unit 21, a system memory 22, and a system bus 23 that operatively couples various system components include the system memory to the processing unit 21. There may be only one or there may be more than one processing unit 21, such that the processor of computer 20 comprises a single central-processing unit (CPU), or a plurality of processing units, commonly referred to as a parallel processing environment. The computer 20 may be a conventional computer, a distributed computer, or any other type of computer; the invention is not so limited.

The system bus 23 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory may also be referred to as simply the memory, and includes read only memory (ROM) 24 and random access memory (RAM) 25. A basic input/output system (BIOS) 26, containing the basic routines that help to transfer information between elements within the computer 20, such as during start-up, is stored in ROM 24. The computer 20 further includes a hard disk drive 27 for reading from and writing to a hard disk, not shown, a magnetic disk drive 28 for reading from or writing to a removable magnetic disk 29, and an optical disk drive 30 for reading from or writing to a removable optical disk 31 such as a CD ROM or other optical media.

The hard disk drive 27, magnetic disk drive 28, and optical disk drive 30 are connected to the system bus 23 by a hard disk drive interface 32, a magnetic disk drive interface 33, and an optical disk drive interface 34, respectively. The drives and their associated computer-readable media provide nonvolatile storage of computer-readable instructions, data structures, program modules and other data for the computer 20. It should be appreciated by those skilled in the art that any type of computer-readable media which can store data that is accessible by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, random access memories (RAMs), read only memories (ROMs), and the like, may be used in the exemplary operating environment.

A number of program modules may be stored on the hard disk, magnetic disk 29, optical disk 31, ROM 24, or RAM 25, including an operating system 35, one or more application programs 36, other program modules 37, and program data 38. A user may enter commands and information into the personal computer 20 through input devices such as a keyboard 40 and pointing device 42. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 21 through a serial port interface 46 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port, or a universal serial bus (USB). A monitor 47 or other type of display device is also connected to the system bus 23 via an interface, such as a video adapter 48. In addition to the monitor, computers typically include other peripheral output devices (not shown), such as speakers and printers.

The computer 20 may operate in a networked environment using logical connections to one or more remote computers, such as remote computer 49. These logical connections are achieved by a communication device coupled to or a part of the computer 20; the invention is not limited to a particular type of communications device. The remote computer 49 may be another computer, a server, a router, a network PC, a client, a peer device or other common network node, and typically includes many or all of the elements described above relative to the computer 20, although only a memory storage device 50 has been illustrated in FIG. 1. The logical connections depicted in FIG. 1 include a local-area network (LAN) 51 and a wide-area network (WAN) 52. Such networking environments are commonplace in office networks, enterprise-wide computer networks, intranets and the Internet, which are all types of networks.

When used in a LAN-networking environment, the computer 20 is connected to the local network 51 through a network interface or adapter 53, which is one type of communications device. When used in a WAN-networking environment, the computer 20 typically includes a modem 54, a type of communications device, or any other type of communications device for establishing communications over the wide area network 52, such as the Internet. The modem 54, which may be internal or external, is connected to the system bus 23 via the serial port interface 46. In a networked environment, program modules depicted relative to the personal computer 20, or portions thereof, may be stored in the remote memory storage device. It is appreciated that the network connections shown are exemplary and other means of and

communications devices for establishing a communications link between the computers may be used.

The hardware and operating environment in conjunction with which embodiments of the invention may be practiced has been described. The computer in conjunction with which embodiments of the invention may be practiced may be a conventional computer, a distributed computer, or any other type of computer; the invention is not so limited. Such a computer typically includes one or more processing units as its processor, and a computer-readable medium such as a memory. The computer may also include a communications device such as a network adapter or a modem, so that it is able to communicatively couple other computers.

System

In this section of the detailed description, a description of a computerized system according to an embodiment of the invention is provided. The description is provided by reference to FIG. 2. Referring now to FIG. 2, a system according to an embodiment of the invention is shown. The system includes a Registry 200, an application 202, and an object 204. The Registry 200 is one type of data store; the invention is not so particularly limited. The Registry 200 is the data store used in conjunction with versions of the Microsoft Windows operating system.

In another embodiment of the invention, at run-time, the object itself determines what behaviors it believes it needs for a given situation, and only instantiates those behaviors for itself. Thus, dynamism in the object behaviors is achieved in two manners: first, the behaviors stored in the object's definition in the data store can be added or removed (viz., changed), and, second, the object is able to determine which behaviors to actually instantiate at run-time.

That is, behaviors may be instantiated as follows: with respect to the definition of the object stored in the Registry, behaviors can be added and subtracted a priori – before the object is ever instantiated. As a result of these newly instantiated objects, they may themselves instantiate additional behaviors – during the object instantiation (although they themselves do not necessarily have to be behaviors). When the object is instantiated, it can determine itself that not all the behaviors need to be included, such that it instantiates only those behaviors for itself that are necessary for a given situation, environment, etc.

This is accomplished in one embodiment as follows:

- 1) an object name is retrieved from the registry;
- 2) the object is instantiated;
- 3) if it has a behavior interface, then such an interface is added directly to the object;
- 4) if it has an add-more-behaviors interface calls the method associated with this interface, passing to the method the object itself;
- 5) the preceding step/act is repeated until there are no more entries; and,
- 6) the object itself is now capable of adding behaviors.

The Registry 206 stores data 206 regarding a plurality of object behaviors for the object 204. These object behaviors are associated with the particular object 204. They may include: other objects, methods, events, properties, etc.; the invention is not so limited. In one embodiment, the object 204 is an object in accordance with the Component Object Model (COM), known within the art, such that the object behaviors are also in accordance with COM. The object behaviors may be system-defined or application-defined. They are dynamic in that behaviors regarding an object, as the data 206 thereregarding is stored in the Registry 200, may be added or removed.

The application 202 is in one embodiment an application program, such as a word processing program, a web browser program, etc. The application 202 as shown in FIG. 2 is instantiating the object 204 by issuing a CoCreateInstance() command, represented by arrow 208 and as known within the art. This results in the instantiation of the object 204, using the object behaviors as the data 206 thereregarding is stored in the registry 200, as represented by arrow 210.

Thus, the invention provides for dynamic object behavior. The object behaviors of the object 204, for example, may be initially system-defined -- i.e., a default set of object behaviors. The application 202, however, may determine prior to instantiating the object 204 that it requires the object 204 to have behaviors that the object does not currently have. Therefore, it may add object behaviors to the data 206 stored in the Registry 200, prior to instantiating the object. This compares with the prior art, which provides for statically defined object behaviors of objects. Furthermore, the invention provides for dynamic object behavior in that the object may itself determine which of the object behaviors to instantiate based on the situation at hand.

It is noted that as to the embodiments of the invention described in conjunction with FIG. 2, the object, the data store, and the application can in one embodiment be stored on a computer-readable storage medium, such as a floppy disk or a compact-disc read-only-memory (CD-ROM). In another embodiment, the objects are such that data representing them resides in a memory of a computer, executed by the processor thereof. The invention is not particularly limited, however.

Methods

In this section of the detailed description, methods according to an embodiment of the invention is presented. This description is provided in reference to FIG. 3. The computerized method is desirably realized at least in part as one or more programs running on a computer -- that is, as a program executed from a computer-readable medium such as a memory by a processor of a computer. The programs are desirably storable on a computer-readable medium such as a floppy disk or a CD-ROM, for distribution and installation and execution on another (suitably equipped) computer. Thus, in one example, an application program is executed by a processor of a computer from a memory thereof to instantiate an object from data stored in a data store of the memory, the data regarding the plurality of dynamic behaviors.

Referring now to FIG. 3, a flowchart of a method according to one embodiment of the invention is shown. In 300, a command to instantiate an object having a plurality of dynamic behaviors is received. This command may be, for example, a CoCreateInstance() command, as known within the art. In 302, data regarding the plurality of dynamic behaviors is looked up in a data store. In one embodiment, the data store may be the Registry, as is used in versions of the Microsoft Windows operating system and known within the art. The behaviors are dynamic in that they may be changed, as will be described. Thus, in 304, the object is instantiated, from the data regarding the plurality of dynamic behaviors of the object stored in the data store.

In 306, the dynamic behaviors of the object are changed. In one embodiment, this includes deleting one of the dynamic behaviors of the object. In another embodiment, this includes adding a new dynamic behavior to the dynamic behaviors of the object. Changing the dynamic behaviors of the object can also mean that the data stored in the data store regarding the plurality of dynamic behaviors is itself changed.

When the object is again instantiated, the object has different behaviors as when previously instantiated in 304. Thus, in 308, data regarding the plurality of dynamic behaviors is looked up in the data store, as those dynamic behaviors have changed. In 310, the object is instantiated, again from the data regarding the plurality of dynamic behaviors stored in the data store, but as has been changed. This means that the object instantiated in 310 has different dynamic object behaviors as compared to the object as instantiated in 304.

It is noted that the invention is not limited to the method of FIG. 3. For example, in the embodiment where the dynamic behaviors are stored in a dynamic store integral to the object itself, there may be many dynamic behaviors that can be instantiated by the object. However, because the object is able to determine which behaviors to actually instantiate – based on the current environment, etc. – dynamic behaviors are provided for in a manner different than that of FIG. 3. In such an embodiment, rather than looking up the data regarding the behaviors in 308, a (second) object is instantiated whose purpose is to provide data regarding the plurality of dynamic behaviors. The object desired is then instantiated.

Conclusion

Dynamic object behavior has been described. Although specific embodiments have been illustrated and described herein, it will be appreciated by those of ordinary skill in the art that any arrangement which is calculated to achieve the same purpose may be substituted for the specific embodiments shown. This application is intended to cover any adaptations or variations of the present invention. Therefore, it is manifestly intended that this invention be limited only by the following claims and equivalents thereof.

We claim:

1. A system comprising:
an object having a plurality of dynamic behaviors;
a data store to store data regarding the plurality of dynamic behaviors; and,
an application to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors,
wherein the object instantiates at least one of the plurality of dynamic behaviors.
2. The system of claim 1, wherein the data store is within the object.
3. The system of claim 2, wherein less than all of the plurality of dynamic behaviors of the object are instantiated.
4. The system of claim 2, wherein the object determines at run-time which of the plurality of dynamic behaviors to instantiate.
5. The system of claim 1, wherein the object comprises a Component Object Model (COM) object.
6. The system of claim 1, wherein the plurality of dynamic behaviors comprises at least one selected from the group essentially consisting of: a plurality of objects, a plurality of methods, and a plurality of events.
7. The system of claim 1, wherein the data store comprises the Registry.
8. The system of claim 1, wherein the plurality of dynamic behaviors comprises a plurality of system-defined behaviors and a plurality of application-defined behaviors.

9. A method comprising:
receiving a command to instantiate an object having a plurality of dynamic behaviors;
looking up data regarding the plurality of dynamic behaviors in a data store; and,
instantiating the object from the data regarding the plurality of dynamic behaviors in the
data store.
10. The method of claim 9, further comprising changing the plurality of dynamic behaviors.
11. The method of claim 10, wherein changing the plurality of dynamic behaviors comprises
deleting one of the plurality of dynamic behaviors.
12. The method of claim 10, wherein changing the plurality of dynamic behaviors comprises
adding a new dynamic behavior to the plurality of dynamic behaviors.
13. The method of claim 10, wherein changing the plurality of dynamic behaviors comprises
changing the data stored in the data store regarding the plurality of dynamic behaviors.
14. The method of claim 10, further comprising:
looking up data regarding the plurality of dynamic behaviors in the data store as have
been changed; and,
instantiating the object from the data regarding the plurality of dynamic behaviors as have
been changed stored in the data store.
15. The method of claim 14, further comprising:
instantiating a second object to provide data regarding the plurality of dynamic behaviors;
and,
instantiating the object from the data regarding the plurality of dynamic behaviors.
16. A computer-readable medium having data stored thereon representing:

an object having a plurality of dynamic behaviors;
a data store to store data regarding the plurality of dynamic behaviors; and,
an application to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors.

17. A computer-readable medium having a computer program stored thereon for execution on a computer, the program performing the method comprising:

receiving a command to instantiate an object having a plurality of dynamic behaviors;
looking up data regarding the plurality of dynamic behaviors in a data store;
instantiating the object from the data regarding the plurality of dynamic behaviors in the data store;
changing the plurality of dynamic behaviors;
looking up data regarding the plurality of dynamic behaviors in the data store as have been changed; and,
instantiating the object from the data regarding the plurality of dynamic behaviors as have been changed stored in the data store.

18. A computer comprising:

a memory;
a processor;
a data store of the memory to store data regarding a plurality of dynamic behaviors of an object; and,
an application executed by the processor from the memory to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors.

19. The computer of claim 14, wherein the object comprises a Component Object Model (COM) object, and the data store comprises the Registry.

ABSTRACT OF THE DISCLOSURE

Dynamic object behavior for object-oriented environments. In one embodiment, a system having an object, a data store, and an application. The object has a plurality of dynamic behaviors. The data store stores data regarding the plurality of dynamic behaviors. The application instantiates the object from data stored in the data store regarding the plurality of dynamic behaviors.

0919604 40966760

"Express Mail" mailing label number: EL185283807US

Date of Deposit: November 25, 1998

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231

Printed Name Chris Hammond

Signature Chris Hammond

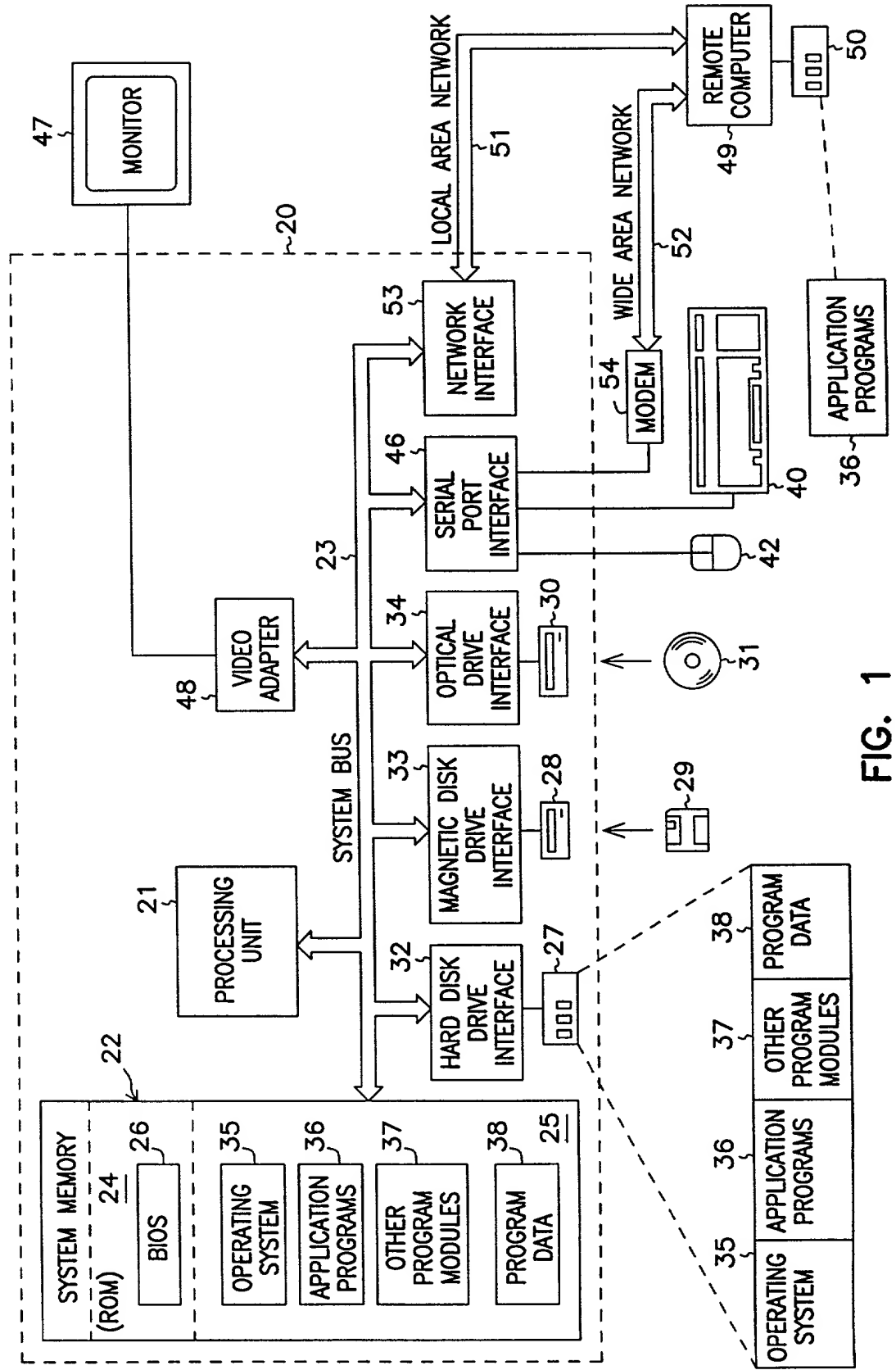


FIG. 1

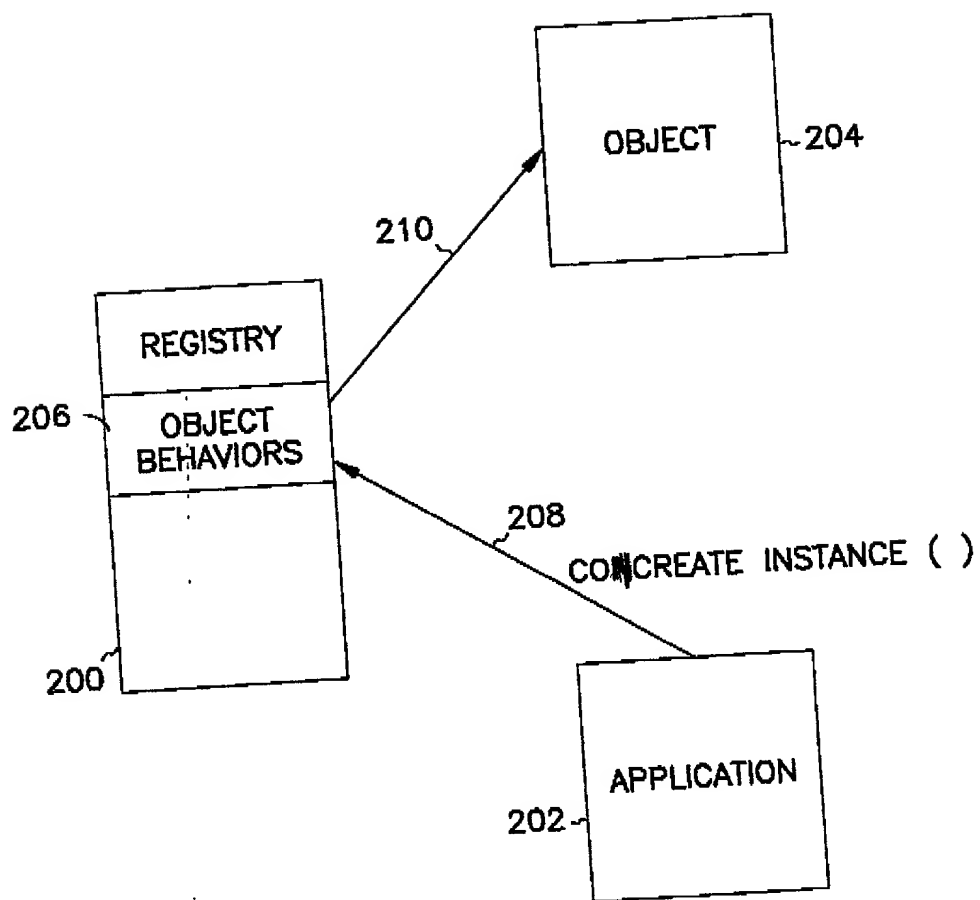


FIG. 2

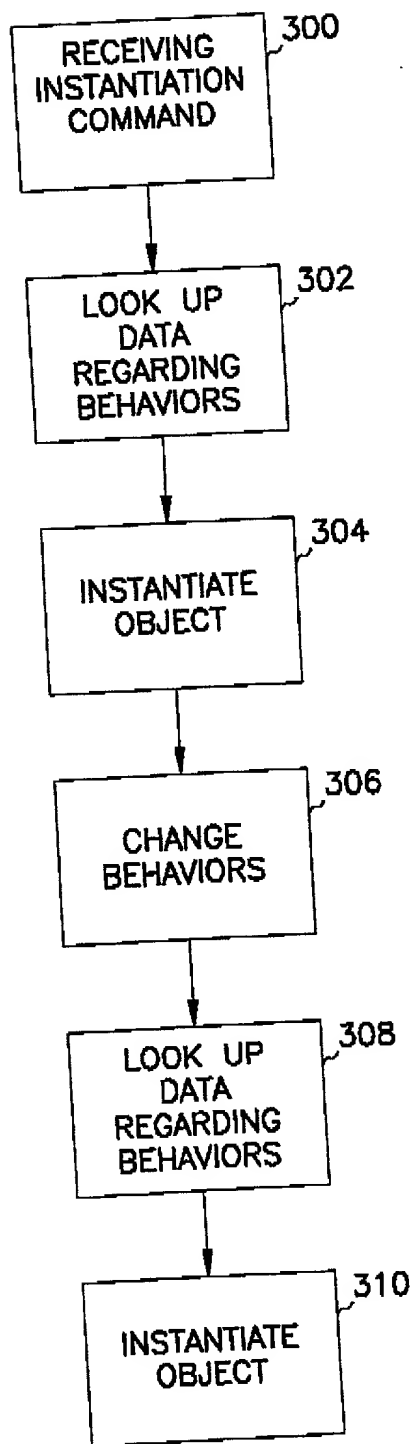


FIG. 3

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.

United States Patent Application

COMBINED DECLARATION AND POWER OF ATTORNEY

As a below named inventor I hereby declare that: my residence, post office address and citizenship are as stated below next to my name; that

I verily believe I am the original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: DYNAMIC OBJECT BEHAVIOR FOR OBJECT-ORIENTED COMPUTING ENVIRONMENTS.

The specification of which is attached hereto.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, § 1.56 (see page 3 attached hereto).

I hereby claim foreign priority benefits under Title 35, United States Code, § 119/365 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on the basis of which priority is claimed:

No such applications have been filed.

I hereby claim the benefit under 35 U.S.C. § 119(e) of any United States provisional application(s) listed below.

No such applications have been filed.

I hereby claim the benefit under Title 35, United States Code, § 120/365 of any United States and PCT international application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, § 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application.

No such applications have been filed.

I hereby appoint the following attorney(s) and/or patent agent(s) to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith:

Aglin, J. Michael	Reg. No. 24,916	Forrest, Bradley A.	Reg. No. 30,837	Lundberg, Steven W.	Reg. No. 30,568
Atora, Suncel	Reg. No. 42,267	Halc, Jeffrey D.	Reg. No. 40,012	Mates, Robert E.	Reg. No. 35,271
Bianchi, Timothy R.	Reg. No. 39,610	Harris, Robert J.	Reg. No. 37,346	McCrackin, Ann M.	Reg. No. 42,858
Billion, Richard E.	Reg. No. 32,836	Holloway, Sheryl S.	Reg. No. 37,850	Padys, Danny J.	Reg. No. 35,635
Black, David W.	Reg. No. 42,331	Huebsch, Joseph C.	Reg. No. 42,673	Polglaze, Daniel J.	Reg. No. 39,801
Brennan, Thomas F.	Reg. No. 35,075	Kalis, Janal M.	Reg. No. 37,650	Sako, Katie E.	Reg. No. 32,628
Brooks, Edward J., III	Reg. No. 40,925	Klima-Silberg, Catherine I.	Reg. No. 40,052	Schwegman, Michael L.	Reg. No. 25,816
Clark, Barbara J.	Reg. No. 38,107	Kluth, Daniel J.	Reg. No. 32,146	Sieffert, Kent J.	Reg. No. 41,312
Crouse, Daniel D.	Reg. No. 32,022	Lacy, Rodney L.	Reg. No. 41,136	Slifer, Russell D.	Reg. No. 39,838
Drake, Eduardo E.	Reg. No. 40,594	Leffert, Thomas W.	Reg. No. 40,697	Terry, Kathleen R.	Reg. No. 31,884
Dryja, Michael A.	Reg. No. 39,662	Lemaire, Charles A.	Reg. No. 36,198	Viksins, Ann S.	Reg. No. 37,748
Embretson, Janet E.	Reg. No. 39,665	Litman, Mark A.	Reg. No. 26,390	Woessner, Warren D.	Reg. No. 30,440
Fogg, David N.	Reg. No. 35,138				

I hereby authorize them to act and rely on instructions from and communicate directly with the person/assignee/attorney/firm/organization/who/which first sends/sent this case to them and by whom/which I hereby declare that I have consented after full disclosure to be represented unless/until I instruct Schwegman, Lundberg, Woessner & Kluth, P.A. to the contrary.

Please direct all correspondence in this case to Schwegman, Lundberg, Woessner & Kluth, P.A. at the address indicated below:

P.O. Box 2938, Minneapolis, MN 55402

Telephone No. (612)373-6900

Our Ref. 777.172US1
Serial No. not assigned
Filing Date: not assigned

Page 2 of 3

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of joint inventor number 1: **David A. Sobeski**Citizenship: **United States of America**Residence: **Redmond, WA**Post Office Address: **15817 NE 90th Street
Apt. E233
Redmond, WA 98052**

Signature: _____

David A. Sobeski

Date: _____

23-Nov-98

Full Name of joint inventor number 2: **Felix G.T.I. Andrew**Citizenship: **Great Britain**Residence: **Seattle, WA**Post Office Address: **5221 37th Avenue NE
Seattle, WA 98105**

Signature: _____

Felix G.T.I. Andrew

Date: _____

20 NOV 98

Full Name of inventor:

Citizenship:

Residence:

Post Office Address:

Signature: _____

Date: _____

Full Name of inventor:

Citizenship:

Residence:

Post Office Address:

Signature: _____

Date: _____

Our Ref. 777.172US1
Serial No. not assigned
Filing Date: not assigned

Page 3 of 3

§ 1.56 Duty to disclose information material to patentability.

(a) A patent by its very nature is affected with a public interest. The public interest is best served, and the most effective patent examination occurs when, at the time an application is being examined, the Office is aware of and evaluates the teachings of all information material to patentability. Each individual associated with the filing and prosecution of a patent application has a duty of candor and good faith in dealing with the Office, which includes a duty to disclose to the Office all information known to that individual to be material to patentability as defined in this section. The duty to disclose information exists with respect to each pending claim until the claim is canceled or withdrawn from consideration, or the application becomes abandoned. Information material to the patentability of a claim that is canceled or withdrawn from consideration need not be submitted if the information is not material to the patentability of any claim remaining under consideration in the application. There is no duty to submit information which is not material to the patentability of any existing claim. The duty to disclose all information known to be material to patentability is deemed to be satisfied if all information known to be material to patentability of any claim issued in a patent was cited by the Office or submitted to the Office in the manner prescribed by §§ 1.97(b)-(d) and 1.98. However, no patent will be granted on an application in connection with which fraud on the Office was practiced or attempted or the duty of disclosure was violated through bad faith or intentional misconduct. The Office encourages applicants to carefully examine:

- (1) prior art cited in search reports of a foreign patent office in a counterpart application, and
- (2) the closest information over which individuals associated with the filing or prosecution of a patent application believe any pending claim patentably defines, to make sure that any material information contained therein is disclosed to the Office.

(b) Under this section, information is material to patentability when it is not cumulative to information already of record or being made of record in the application, and

- (1) It establishes, by itself or in combination with other information, a prima facie case of unpatentability of a claim; or
- (2) It refutes, or is inconsistent with, a position the applicant takes in:
 - (i) Opposing an argument of unpatentability relied on by the Office, or
 - (ii) Asserting an argument of patentability.

A prima facie case of unpatentability is established when the information compels a conclusion that a claim is unpatentable under the preponderance of evidence, burden-of-proof standard, giving each term in the claim its broadest reasonable construction consistent with the specification, and before any consideration is given to evidence which may be submitted in an attempt to establish a contrary conclusion of patentability.

(c) Individuals associated with the filing or prosecution of a patent application within the meaning of this section are:

- (1) Each inventor named in the application;
- (2) Each attorney or agent who prepares or prosecutes the application; and
- (3) Every other person who is substantively involved in the preparation or prosecution of the application and who is associated with the inventor, with the assignee or with anyone to whom there is an obligation to assign the application.

(d) Individuals other than the attorney, agent or inventor may comply with this section by disclosing information to the attorney, agent, or inventor.